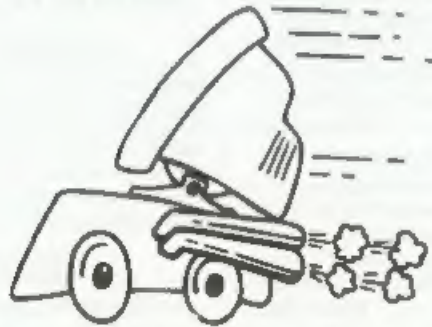


BASIC programs run too slow?
Machine Code too hard to write?



The language all the professionals use...
Almost as easy as writing in BASIC...
Yet it compiles to Machine Code...

SAM C

Portable code - transfer C source files to and from PC's and Amiga's.

Standard libraries, technical documentation and beginner's guide to C all included in package.

Compiled code can be commercially released!

Release Date : 29/4/95, Price : £19.99 - send SAE for full list of features and capabilities.



FRED PUBLISHING, 40 ROUNDYHILL, MONIFIETH, DUNDEE, DD5 4RZ.
Tel: 01382 535963

Vol.8 N°12.

August 1995.

FORMAT

THE HOME OF Z80 COMPUTING

AGENT MULDER BELIEVES HIS FORMAT HAS BEEN STOLEN BY ALIENS... AGENT SCULLY REMAINS SCEPTICAL.



Make Sure You Keep
Your *FORMATs* Safe

£1.50 (UK R.R.P.)

ISSN 0963-8598.

B.G.SERVICES

64, Roebuck Road, Chessington, Surrey, KT9 1JX. Tel: 0181-287-4180.

SPECTRUM PRODUCTS

- +2A MOD KIT** Converts a black +2A into a genuine +2 with standard expansion port. This allows the PLUS D to operate properly. Includes replacement ROM, FIXER and full fitting instructions + manual updates. **£15.50.**
- THE FIXIT** Converts the +3 and +2a expansion port so that some 48K add-ons will work. Single port version **£9.50**, double port version **£10.50.**
- VTX5000** **SPECIAL OFFER** - V23 direct connect modem with viewdata software in ROM. Version for 48K or gray +2 (+2A/+3 users need FIXIT) - **£10.**
- PD TAPE 01** Comms software for VTX5000, VTX711 and RS232s. **ONLY £1.50.**
- Dr. SCROLL** Scrolling software for VTX5000. **£4.00.**
- MEMBRANES** Spectrum 48K (rubber type) keyboard membranes. **£6.00.**
Spectrum+ / 128K keyboard membranes **£15.00.** *While Stocks Last*
- +3 PD TAPE** +3 Utilities. Side 1 +3DOS. Side 2 CPM files. **£1.50.**
- THE RITMAN COLLECTION** Four classic games on one DISCiPLE/PLUS D disc. Bear Bover, Combat Zone, Cosmic Debris and Dimension Destructors. **ONLY £5.00**
Also works with registered version of Z80 Spectrum Emulator on IBM.
- The Last Word** A full featured word-processor for the Spectrum. Includes many unique features. Supplied on tape for easy conversion to disc. **ONLY £9.50**
**** NEW ****

SAM PRODUCTS

- PRO-DOS LITE** Allows you to run CP/M 2.2 programs on SAM, supplied with full manual on disc and over 1Mb PD software. **Special Trial Offer £12**
- CP/M MANUALS** Genuine CP/M 2.2 DR Manuals **£6.50**

□ TURN YOUR PC INTO A SPECTRUM □

Z80 - New version (V3.0) of the best Spectrum emulator for your IBM compatible. Full Registered version **£20.** (version without DISCiPLE/PLUS D emulation **£15**).
Upgrades **£10** (full) **£5** (non-DISCiPLE/PLUS D) to registered users.
Tape interface, ready built, plugs into parallel port, **£14.95** incl p&p.

NEW - Mind Games disc for Z80 users. - **NEW**

Contains Rebelstar, Rebelstar II, Chaos, Nebula, Lasersquad and Bounty.
With instructions on disc. Need Z80 to run (see above) Only **£5.**

□ RIBBON RE-INKING SERVICE □

Star, Citizen and other standard fabric ribbons re-inked for **£2** each + return postage.

Please allow 28 days for delivery. All prices include UK postage and packing.
Overseas customers please add £2 on modems, PRO-DOS and +3 CP/M, add £1 on all other items.
Please make cheques/postal orders payable to B.G. Services.

CONTENTS

Vol.8 N°12.

August 1995.

- ♦ News On 4 4
- ♦ The Editor Speaks 5
- ♦ Short Spot 7
- ♦ Memory Is Made Like This 13
- ♦ How Basic Is Stored 19
- ♦ Exploring SAM MIDI 23
- ♦ Variables On A Theme - Part 3 28
- ♦ Your Letters 31
- ♦ **FORMAT** Readers Service 39

STOP PRESS: I'm sorry to say that due to last second production problems this month's episode of the Fastfile series has been delayed until next month.

THIS MONTHS ADVERTISERS:-

ATOMIC SOFTWARE	18
B.G.SERVICES	2
CHEZRON SOFTWARE	30
FORMAT (SAM REPAIR SERVICE)	11
FRED PUBLISHING	Back Cover
HILTON COMPUTER SERVICES	34
REVELATION SOFTWARE	22
S.D.SOFTWARE	6
STEVE'S SOFTWARE	17
WEST COAST MAIL ORDER	12

FORMAT ISSN 0963-8598

© Copyright 1995 **FORMAT PUBLICATIONS.** All Rights Reserved.

No part of this publication may be reproduced, in any form, without the written consent of the publisher. The purchaser of this magazine may copy program material published herein for his or her own personal use and provided that any such programs are erased if and when the purchaser disposes of this magazine in any way. While every effort is made to ensure accuracy in **FORMAT** the publisher will not be held liable for any errors or omissions.

FORMAT is published by **FORMAT PUBLICATIONS.**

34, Bourton Road, Gloucester, GL4 0LE, England.

Telephone 01452-412572. Fax 01452-380890.

Printed by D.S.Lilho, Gloucester, Telephone 01452-523198.

NEWS ON 4

NEW DATA STORAGE FOR SAM

Exeter based Revelation Software have announced a new data-storage program for SAM called *PROFILE*. The system is capable of storing up to 999,999 records and works in a very similar fashion to a card file.

Designed to be expanded by the user, *PROFILE* is being projected as the answer to virtually any data storage problem SAM users can come up with. Revelation are also hinting that a Spectrum version may be ready for the New Year.

Revelation hope to have *PROFILE* ready to launch at the October show in Gloucester at a very reasonable £14.95. We hope to bring you a review soon.

QUAZAR SHIPPING

As an update to our news item in the May issue we are now pleased to say that the new sound card for SAM - The Quazar Surround - is now in production and available to buy from Colin Piggot, 204 Lamond Drive, St. Andrews, Fife, KY16 8RR.

The system costs £61.99 and offers 6 channel sound in up to 16 bit resolution. Readers can send an SAE for further information and we hope to arrange a review for *FORMAT* as soon as we can borrow a unit from Colin.

While on the subject of Quazar, Jupiter Software have released the first game to make full use of the Quazar Surround board. Called *Amalthea* the game will play on a standard SAM but comes into its own with the Quazar system attached. Boasting 800K of sampled sounds *Amalthea* comes on three discs and is available for just £10 from Jupiter

Software 2, Oswald Road, Rushden, Northants, NN10 0LE. (Make cheques payable to S.Ekins please).

ULTRA CHEAP KEYBOARD

The latest sales list from Greenweld Electronics of Southampton shows a Spectrum +3 keyboard assembly for the unbelievable price of £1.00.

Described as the Spanish version there may be a few keytop differences but at that price it is worth it just for the membrane. Contact Greenweld on 01703 236363 for more details on ordering, quote part number Z9210.

NINTENDO CUTS UK OPERATION

Console veteran Nintendo is to drastically cut its UK workforce over the next two months as a step towards a new distribution set-up in time for the Christmas market. They hope to redeploy staff but there may be redundancies.

YOUR NEWS

Obviously this is a bad time of the year for computer news. But this space could well have been filled with your news, if only you had sent it to us. There must be local computer clubs out there that want publicity, or perhaps you have heard of a new printer that has just hit the market.

Remember, you get three months extra subscription for every news item we print, so don't rely on others to send in an item you have seen - send it off yourself and earn our thanks as well.

Credits: Ken Todd.

URGENT we need your news. Anything you think other people should know about. Each item printed earns the contributor 3 months extra subscription (please claim when next renewing).

The Editor Speaks

If I thought it was hot when I wrote last month's editorial then it is just about impossible as I write this one. Neither I, nor the equipment I work with, like the heat (and to give you some idea it was 102°F on Saturday in the office and it has been in the 90's every day this week). So if it turns out there are any errors in this month's issue, sorry, my brain is just too hot to work properly.

Right, first this month is the important news that the next Gloucester show has been fixed for the 14th October. On a Saturday as usual, which the majority of you seem to want us to continue with although there has been some call for Sundays, and at the same venue as the others we have done. We look forward to seeing lots of familiar faces and, we hope, many new ones on the day.

This time we are hoping to arrange a shuttle service to get people from the railway/bus station in the centre of town out to the show. However, we have to work out how many people would need such a service before we commit to it. So, if you think you will be coming by train or coach, then please drop us a quick line right away so that we can judge what the demand will be and sort out details.

Judging from the letters we get, and we do get far more than we can ever publish in the letters page, there is a very big demand for more tutorials like Carol's acclaimed 'Machine Code Without The Tears' series, or 'Basically Speaking' (which we hope will return in the next few months). We have had requests for C of course, as that seems to be the 'in' language for SAM at the moment (if you have not got your copy

yet then order it now because it is well worth the money) and of course C is available on the Spectrum by HiSoft. There is also interest in FORTH (although we did do a series on that in the early years of *FORMAT*), PASCAL, COBOL (horrid language I assure you), FORTRAN and others. If anyone out there in reader's-land knows a bit about one of these then how about passing on your knowledge for the benefit of others. There were lots of languages produced for the Spectrum, but even if the language you know does not have a compiler at the moment on the Spectrum or SAM then telling others about the language may generate enough interest for a compiler to be written.

Right, staying on the same subject, I now have a little appeal. I've lost a name and address. Silly me. Some time ago there was a company called MIRA (I think) who produced Pascal and Fortran compilers for the Spectrum. If anyone can let me have an address and if possible a telephone number I would be most grateful.

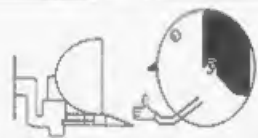
Now just a little reminder. Last month I asked for as many letters as possible on the general line of 'How I got Involved With Computers' or 'How I Came To Own A Speccy/SAM'. We have had some already, but I still want more, so get writing and send your letter off very quickly so we stand a chance of including it in next month's issue.

Oh well, that's it for another month, hopefully the office will cool down a bit next week so I can get cracking on a super birthday issue for you.

Until next month.

Bob Brenchley, Editor.

S D S SOFTWARE



Quality Serious Software For Spectrum & SAM

PC-SUITE

From the same author as SPECMAKER. Now you can transfer your IBM data files by reading and writing IBM discs (720K format) on your SAM. PC-SUITE will even let you format IBM discs on your SAM. Write and edit SAM Basic programs on your PC. Use PC-SUITE to copy SAM data files to PC so you can print them on that laser printer at work... Supplied on 3 1/2" disc.

As used by Format Publications to transfer articles and programs for this mag.

SPECMAKER

The simplest Spectrum emulator on the market for SAM. Thousands of 48K programs work without the need for any conversion. Most other programs need only minor changes. All the extra SAM keys work in Spectrum mode. Uses SAM's printer port and up to 390K of SAM's memory as a RAMDISC. PLUS D, DISCIPLE and OPUS+ discs can be loaded into SPECMAKER and saved to SAM disc. Converts files between Messenger and SPECMAKER format to save valuable disc space. Comes with pre-converted Spectrum ROM image (no need to own a Spectrum). Supplied on 3 1/2" disc.

**Requires MasterDisc and MasterBasic to use OPUS single density discs.*

Spectrum Owners

Don't forget to say if ordering for PLUS D or DISCIPLE and state disc size - 3 1/2" or 5 1/4" (80 or 40 Tracks).

PLUS MEMBERS

Don't forget your membership number if claiming a discount.

	Prices: Non Members	PLUS Members	Available for: DISCIPLE PLUS D	GOOD	UNIDOS	SAM	3 1/2"	5 1/4"	5 1/4" Tapes
The Fractal Collection	£5.95	£4.95	✓	✓	✓	✓	✓	✓	✓
Specfile+	£12.95	£9.95	✓	✓	✓	✓	✓	✓	✓
Uni-Dos	£25.95	£19.95	✓	✓	✓	✓	✓	✓	✓
Createfile Manual	£8.50	£5.50	✓	✓	✓	✓	✓	✓	✓
SpecMaker	£12.95	£9.95	✓	✓	✓	✓	✓	✓	✓
PC Suite	£25.95	£19.95	✓	✓	✓	✓	✓	✓	✓
MIDI+SOUND	£25.00	£22.50	✓	✓	✓	✓	✓	✓	✓
Hackers Workbench	£9.90	£8.90	✓	✓	✓	✓	✓	✓	✓
ISU	£4.90	£3.50	✓	✓	✓	✓	✓	✓	✓
DBU	£5.50	£4.00	✓	✓	✓	✓	✓	✓	✓
File Converter	£5.30	£4.50	✓	✓	✓	✓	✓	✓	✓

Please add £2.00 to price for each 3" disc required.

Please add £0.50 postage for each order (£1.20 outside UK).

* Using SpecMaker Spectrum emulator

**S. D. Software, 70, Rainhall Road,
Barnoldswick, Lancashire, England, BB8 6AB.**

SHORT SPOT

YOUR HINTS, TIPS AND PROGRAMMING IDEAS

Edited By:- John Wass.

Incredibly, Short Spot time is upon me again and I've only just got back from the States. Anyway, as luck will have it, I managed to get most of this episode typed in before I flew off... So here goes!

Firstly, as a followup to May's Short Spot (the bit with notes on Two Ants and the comments on SAM's cursor keys) there's a couple of letters from John Saunders of Chalfont St. Giles, Bucks, who mentions that Duncan's surname is MUNRO, and that, as in Allan Clarkson's case, failure of the cursor keys has been followed by failure of the CNTRL key, and so the Short Spot SAM program to get round the initial problem is now useless. John agrees with Ray Bray over this - he reckons the behaviour pattern is derived from insecure connections. In this case, there ought to be a simple fix; anyone know what it is?

While I'm at it, I've also now found John's letter of 8th May. In this he mentions that it is strange, how one keeps encountering seemingly obscure puzzles with SAM which are not easily chivvied out with the aid of the manual. The problem is that he was hoping to make use of a particularly long DIM statement, e.g. a\$(7,7,400) to do with some recording of data, and, not wanting acres of blank screen staring out at him, decided to retrieve with TRUNC\$. We are all familiar with the error statement 'Invalid Argument' and very frustrating it is, too, but why here? Well, by experimenting with a short program, John found that the limit of TRUNC\$

seems to be 255! Above that figure, it won't truncate. The following few lines tell the sorry tale...

```
10 FOR b=0 TO 18: PRINT TAB
15,"****"; NEXT b
20 FOR n=240 TO 260
30 DIM A$(4,4,N)
40 LET A$(1,1,1 TO n)="MAYBEN
OT"
50 IF n<242
60 PRINT AT n-240,0;n;" ";
A$(1,1)
70 ELSE PRINT AT n-240,0;n;"
"; TRUNC$ A$(1,1)
80 END IF
90 NEXT n
```

Many thanks, John.

Next we turn to Paul Farrow of South Tottenham, London. Paul has sent in a whole raft of bits and pieces, and it will take a fair chunk of Short Spot to do them all. There's also the further problem that they're mostly Spectrum oriented. Nevertheless, I think the items are of general interest, and the piece of paper's been hanging around for some time. So although last month was mainly a Spectrum issue, we can, I think, mention some of the 'items' without getting too much hate mail. First, some general tips.

Computer Manuals at 50, James Road, Birmingham, B11 2BA, still sell a few Spectrum books, including Toni Baker's 'Mastering Machine Code' priced at £9.95.

Remember the comments about the Spectrum Keypad? Well, Paul has actually used the Spanish version of the Keypad and has examined both the hardware and the software components.

The PIC used in the Keypad has almost certainly had its security fuse blown; hence its code cannot be read out. However, since the Keypad is controlled via RS232, it would be fairly easy to produce a compatible keypad by writing the Comm. software from scratch. However, Paul writes that having used the Keypad, he can't see any benefit in making one, since the functions provided are not all that useful, anyway. And in any case, all the functions available from the Keypad are also accessible from the normal keyboard. This was pointed out by Toni Baker in an issue of ZX computing. A list of the functions which are provided and the method of access from the keyboard follows:-

Move the cursor as far as the beginning of the next word - PRESS EXTND MODE THEN SYMBOL SHIFT J

Move the cursor as far as the next space to the left - PRESS EXTND MODE THEN I

Move the cursor 10 lines up the program - PRESS EXTND MODE THEN P

Move the cursor 10 lines down the program - PRESS SYMBOL SHIFT I

Move the cursor to the end of the Basic line - PRESS EXTND MODE THEN N

Move the cursor to the start of the Basic line - PRESS EXTND MODE THEN SYMBOL SHIFT 2

Move the cursor to the start of the Basic program - PRESS EXTND MODE THEN N

Move the cursor to the end of the Basic program - PRESS EXTND MODE THEN T

Delete the character which the cursor is on - PRESS EXTND MODE THEN SYMBOL SHIFT K

Delete as far as the next space to the left of the cursor - PRESS EXTND MODE THEN E

Delete from the character under the cursor as far as the next space to the

right - PRESS EXTND MODE THEN W

Delete from the left of the cursor to the beginning of the line - PRESS EXTND MODE THEN K

Delete from the character under the cursor to the end of the line - PRESS EXTND MODE THEN J

TOGGLE Switch between upper and lower screen - PRESS EXTND MODE THEN SYMBOL SHIFT B

OK, Paul; nice one. I agree that the functions are all available from the main keyboard, but single-key presses for these would be rather nice, wouldn't they. And keypads are available; for instance as addons for notebooks. Well; challenge of the month: who's going to make a batch for us, then?

Now back to SAM for a minute. I've had a rather sad letter from Alan Cox, who has given his Spectrum hardware and software to charity and put his SAM up for sale. One of the things he has been doing lately is to play with various bits of old computer machinery, including a Data General Enterprise, an ICL DRS20 system, and an Apple IIe, an Acorn Atom and a Hewlett-Packard HP9816. In the manual for the HP, Alan came across a program which simulated a meter; a panel meter, including the needle-shudder one so often gets with such meters. Alan thought at first of an Ammeter, then, by an inexplicable metal leap, of a SAMmeter, and then, of course, he had to go ahead and reproduce the program for SAM, so here it is, warts and all:-

```

10 REM program to reproduce t
   he meter HP program
20 PLOT 0,0: DRAW 255,0: DRAW
   0,150: DRAW -255,0: DRAW
   0,-150: REM draw part fram
   e
30 PLOT 0,20: DRAW 255,0: REM
   complete frame
40 PRINT AT 18,7;"(S) A M M E
   T E R"
50 PRINT AT 0,0;"Use cursor k

```

```

eys to alter reading"
60 scales
70 LET z=0
80 LET z=z+(INKEY$=CHR$( 9))
  -(INKEY$=CHR$( 8))
90 LET i=z+RND
95 IF i>40 THEN LET i=40
96 IF i<-40 THEN LET i=-40
100 PLOT 124,20
110 DRAW 95*SIN (i*PI/180),95
  *COS (i*PI/180)
120 PLOT 124,20
130 DRAW OVER 1,95*SIN (i*PI/1
  80),95*COS (i*PI/180)
140 GOTO 80
150 STOP
160 DEF PROC scales
170 FOR z=-40 TO 40 STEP 10
180 LET zz=z*PI/180
190 LET ss=SIN zz,cc=COS zz
200 PLOT 124,20
210 DRAW PEN 0;100*ss,100*cc:
  DRAW PEN 7;10*ss,10*cc
220 NEXT z
230 PRINT AT 4,15;"0"; AT 5,20
  ;"10"; AT 5,8;"-10"; AT 7,
  24;"20"; AT 7,4;"-20"
240 END PROC
250 DEF PROC sv: SAVE "sammete
  r": END PROC

```

Many thanks, Alan; glad to hear that you have not deserted us completely.

Now over to Roy Burford of Norton, Stourbridge, who, like me, is involved in Education. He writes about a Spectrum program he has been playing with for some time, which is called Rounding (and so has his Commodore 64 mate). Those of us in the Business will know well the problems of handling, manipulating and rounding large numbers of assorted examination and coursework marks, particularly at this time of the year, and this program is supposed to help with such jobs.

The problem was that the original program Roy wrote accepted approximations which were technically wrong, and got the ultimate answers wrong, too, and students object to this!

The following improvements have helped matters

i) The original multiplier, m, was fixed at 10^6 , which caused the program to neglect low-value roundings.

ii) the original determination of the decimal places, d, overlooked the disappearance of the leading cipher for decimals below 0.1, hence the routine 129-135.

iii) The original single value for ch did not work equally well for sf's and dp's.

iv) To stop the program from accepting approximations (e.g. 29.29999 for 29.3), the string lengths were checked on arrival at count. Incidentally, arrival at count directly using LEN STR\$ a - LEN STR\$ ans produced an incorrect result, no matter how the expression was bracketted!

Roy writes that he is still not 100% sure of the integrity of this program, but it is the best to date. He also writes that he firmly believes, now, that the number-holding accuracy of the Spectrum is inadequate for the accuracy demands of the program.

Well, here it is:-

```

1 REM Wain & Flower. Mathemat
  ics Homework on a Micro. Un
  iversaty Printing Service(L
  eeds). W11. Rounding.
2 REM Revised to ZX Spectrum+
  128K by B.C.R.Burford 2005
  94.
3 REM Second revision 250794.
  3rd 260794(minor). 4th 270
  894. 5th 281294.
30 PRINT AT 10,0;"          Round
  ing""
40 PRINT "Enter -1 to quit ear
  ly""
45 PRINT "Multiplier?(Integer
  1/1000000) ""';
50 INPUT m: LET m=INT (m)
60 IF m<1 OR m>1E+6 THEN GOTO
  50
65 PRINT "Multiplier= ";m''
68 PRINT "No. problems? ";
70 INPUT q
80 LET q=INT q
85 PRINT q''
87 PAUSE 50
90 IF q=0 THEN LET c=0: LET t=

```

```

0: GOTO 250
100 CLS : LET c=0: LET t=0
110 LET c=c+1: LET n=RND*m: LET
z=RND
120 LET l=LEN STR$ n
125 IF n<l AND n>=.1 THEN LET l
=l-1
127 LET d=0
129 FOR i=LEN STR$ n TO 1 STEP
-1
131 IF CODE (STR$ n)(i)<>46 THE
N GOTO 135
133 LET d=(LEN STR$ n)-i: LET i
=i-1
135 NEXT i
150 LET p=INT (RND*d): IF p<l T
HEN LET p=p+1
152 LET sf=p+1-d-1: IF sf<l THE
N LET sf=sf+1
155 PRINT AT 10,0;"Q ";c;"/";q;
" Multiplier=";m;"
157 PRINT "-1 to quit"
160 PRINT "What is ";n;" rounde
d to "
170 IF s>.5 THEN PRINT 'p;" dec
imal places? ";: LET ch=.01
6: GOTO 190
180 PRINT 'sf;" significant fig
ures? ";: LET ch=.033
190 INPUT a: PRINT a
200 IF a=-1 THEN GOTO 250
210 LET r=INT (n*10^p+.555): LE
T s=a*10^p: LET ans=r/10^p:
LET e=LEN STR$ a: LET f=LE
N STR$ ans: LET count=e-f
212 LET chab=ABS (r-s)
220 IF chab<ch AND count=0 THEN
PRINT "Good.": LET t=t+
1: GOTO 235
230 PRINT "Answer is ";ans;"
235 IF c=q THEN GOTO 250
240 PRINT "Press any key:": PAU
SE 0: IF c<q THEN CLS : GOT
O 110
250 PRINT "You got ";t;" out o
f ";c;" correct."
260 STOP

```

Well, IS the number-holding accuracy of the Spectrum inadequate? What do you think.

Many thanks, Roy.

You know, I know that this is a bit of a lottery, but the beauty of Spectrum Basic programs is that provided they don't involve PEEK and POKE, they'll run without any modification on SAM. What,

too fast? Nonsense - you'll get the result a bit quicker, that's all. And here's where it counts - when it really is a lottery - and you want numbers for it. Yes, folks, once again, Format presents yet another variation, this time by Gerald Symes of Easton, Winchester. And he's nearly got it within the 11 lines I suggested, too.

```

10 REM lottery by G.J.Symes
20 DIM a$(3): GOSUB 120: LET n
$=a$
30 FOR c=1 TO 5
40 GOSUB 120
50 FOR n=1 TO LEN n$ STEP 3
60 IF a$=n$(n TO n+2) THEN GOS
UB 120
70 IF a$>n$(n TO n+2) THEN NEX
T n
80 LET n$=n$ ( TO n-1)+a$+n$(n
TO )
90 NEXT C
100 PRINT "Your lottery nu
mbers are:" " ";n$
110 GOTO 20: REM or STOP
120 LET a$=STR$ INT (RND*49+1)
130 IF a$(2)=" " THEN LET a$="0
"+a$
140 RETURN

```

And may you all have many happy numbers. Many thanks, Gerald.

Now back to SAM for a while. You know, we don't half have trouble timetableing our Pilot Plant Week, which I always run with our MSc students. The problem is a religious one, or so I've been led to believe, and in our School of Chemical Engineering, there's an enormous collection of practising Christians (to say nothing of those of other faiths, too).

We've got lay readers, preachers, a choirmaster: we've even a fully-featured honorary curate! Some years ago, I was intemperante enough to ask why we'd got so many - only the Theology Department seemed to beat us, and after all, they ARE the professionals. Old Uncle Miskin (now retired) bristled. "Easy John," he said. "Chemical Engineers are used to taking imperfect systems - and making

them work!"

But even then, none of them can explain to me quite why Jesus Christ got himself born on the same day each year, but died on quite a different one! I tell you what, though: it doesn't half play the bear with the timetable. Pilot Plant week is, of course, outside the normal undergraduate term, as it needs considerable resources, and the MSc course students are there for a 12-month year. (Take note, those who labour under the misapprehension that the academic's time is pegged to undergraduates: it's not). So sometimes, it's just before Easter and sometimes just after. Makes arranging even a short holiday difficult. Now, As you noticed previously, the Gregorian Calendar is very difficult, with no leap year calculation if the year ends in 00. Pope Gregory's systems analysts must have cursed him for setting them so awkward a task. However, the Julian Calendar, with its simple leap year every four years, set up by the Council of Nicea in AD 325 is much simpler. It would have been very nice had the Julian Calendar been right. It would have been nicer still had Easter not wandered round the calendar (and I can't for the life of me see why it should). Never mind: here is a Spectrum program to give Easter from AD 325 to AD 752, culled from one of the discs that's done the rounds, but again originated from our old friend Ettrick Thomson. Funny: he usually writes for SAM...

```

10 DEF FN M(p,q)=p-q*INT (p/q
|
20 INPUT "Year (<1753):";y
30 PRINT "Easter in ";y;" Feb
l on ";
40 LET dgn=FN M(16+19*FN M(y,
19),30)
50 LET eac=dgn+FN M(-dgn-INT
(1.25*y),7)
60 IF eac<11 THEN PRINT "Marc
h ";eac+21
70 IF eac>10 THEN PRINT "Apri

```

```

1 ";eac-10
80 GOTO 20

```

Many thanks, Ettrick.

And that's all I'm able to do this month, for it takes quite a long time to sort out, duplicate and type up Short Spot, and time's caught up on me. I must go off to unpack my suitcase, mustn't I...

I really need some more snippets, both for the Spectrum and SAM, as I've ground well down the pile. Please, as usual, send them to:-

**John Wase,
Green Leys Cottage,
Bishampton,
Pershore,
Worce,
WR10 2LX.**

Oh yes, please hurry up with some nice Christmassy bits as I have no doubt that Bob will soon be on the phone pestering me for a nice seasonal Short Spot for the December issue.

FORMAT

SAM REPAIR SERVICE

We are pleased to be able to offer readers an official, West Coast approved, repair service for SAM Coupé and SAM élite computers and add-ons.

PRICING

Because every fault is different we don't think it is fair to charge a fixed rate. When we receive a machine it is tested and a quote is sent to you for the repair. If you decide for some reason not to go ahead with the repair you are under no obligation.

Pack your machine well (we will use your packing for its return). Please phone us on 01452-412572 for advice on dispatch (we can even arrange collection). In many cases we can also give an estimate over the phone. All repairs returned by overnight carrier so tell us which days there will be someone in to sign for the parcel. Please quote your INDUG membership number and give a evening contact phone number.

Send your machine to -

**Format (SAM Repairs),
34, Bourton Rd, Gloucester, GL4 0LE.**

Below you will find details of the full range of SAM hardware now available.

CODE	DESCRIPTION	PRICE
COMPUTERS - Now with extended two year warranty.		
W1010	SAM <i>486</i> - 512K - Single Drive. The latest computer in the SAM family. 512K RAM, Single Drive Machine now with Built-in Printer Interface.	£199.95
W1011	SAM <i>486</i> - 512K - As above but with Dual Drives fitted.	£259.95
W1020	SAM <i>486</i> - 512K - Single Drive. As W1010 except we supply scart to scart lead instead of TV modulator so you get the best possible picture from you SAM.	£199.95
W1021	SAM <i>486</i> - 512K - Dual Drive. As W1011 except we supply scart to scart lead instead of TV modulator.	£259.95
INTERFACES (work with both SAM <i>486</i> and SAM Coupé unless otherwise stated)		
W1101	SAM Disc Drive Upgrade. Have a second drive fitted to your SAM <i>486</i> or a replacement/second drive for your Coupé. Fitting and return postage included in price. Ring 01452-412572 for instructions on how to send your computer.	£79.95
W1121	SAM Coupé Disc Upgrade Kit - all you need to upgrade to a twin drive machine. Needs good soldering skills to fit. You also need de-soldering tools to remove one of the connectors on the main SAM board. Comes complete with all instructions, components and a top quality 3 1/2" drive mechanism.	£64.95
W1122	SAM Coupé Disc Upgrade Kit - as above but without drive mechanism.	£29.95
W1102	External Parallel Printer Interface - the SPI. (for Coupé)	£24.95
W1103	RS232/Parallel Interface (COMMS). (for Coupé)	£34.95
W1104	1 MegaByte External Memory Pack. (Needs MasterDOS to run from Basic).	PHONE
W1106	SAM Mouse System. Interface, Mouse and Software. Plugs into mouse port on rear of SAM. More and more software now works better if you use a mouse.	£39.95
W1107	External Disc Drive Interface and Parallel Printer Port (for Coupé)	£44.95
W1108	Internal 256K Upgrade (for older SAMs with only 128K of memory)	£34.95
W1109	<i>TWO!</i> allows 2 interfaces to be connected to SAM expansion port at same time.	£19.95

Prices valid until 1st January 1986. E. & O.E.

Ordering: Write your order clearly, state your name, full address and phone number and quote your INDUG number (if you are a member). Payment. Cheques and UK Postal Orders, should be made payable to **FORMAT PUBLICATIONS.**

Postage & Packing: Price includes UK P&P. European customers please add 15% for airmail delivery, other overseas customers please write for quote. Overseas customers can also pay using the FORMAT foreign exchange scheme.

COMING SOON

A new Clock/Calendar system for SAM, fully compatible with MasterDOS.
And our First Spectrum Interface. More news as soon as we have it.

MEMORY IS MADE LIKE THIS

By:- Dick Guy.

In this brief text I will try to give readers an introduction to computer memory, essential for every type of computer. You may have the fastest microprocessor in the world, but without memory, it's totally useless. Conversely, you do not need a microprocessor to use memory chips, even though applications will be very limited.

An in depth description of memory is beyond the scope of this introduction, so, as far as possible, I have simplified things (as much as I can) by the use of switches to represent transistors (where used). I have also endeavoured to do away with timing diagrams, which are essential if a full understanding of the subject is needed but would cloud the issue if all you want is an understanding of what memory is. I hope those of you out there who understand these things will bear with me, for the benefit of our less knowledgeable colleagues.

There are two major types of 'solid state' memory used in a computer. (There are other types*of course, but I will not confuse things by considering them here). These are RAM, standing for 'Random Access Read/Write Memory', and ROM, standing for 'Random Access Read Only Memory'.

By far the most interesting - and useful is RAM. ROM is also useful and interesting of course, but specialised equipment is needed in most cases, to enable the full potential to be exploited. Most of you will only meet ROM when

you change the operating system of your Speccy, SAM, DISCiPLE, PLUS D or similar, as this is the bit that enables your computer to 'boot up'. To narrow things down a little more therefore, we will concentrate on RAM.

There are now a few 'buzz' words we need to learn about RAM. The first, is RAM is 'volatile'. This means that if we turn off the power we lose any program contained in our computer's memory (A major advantage of ROM is that we do not). The loss of one half cycle of mains supply (approximately 10 one thousandths of a second), is sufficient to totally wipe a RAM memory, unless precautions are included in the design of the computer. (This gets us into another field - that of power supply design, so let us leave the subject there).

The next are related. RAM comes in two principle forms, STATIC and DYNAMIC. They both have strengths and weaknesses, and I will expand on them as we progress.

Static memory was the first type of solid state memory to be used. It is based on a two transistor circuit configuration, called a 'flip-flop'. The name is very descriptive of circuit operation, as by suitable triggering it can be arranged that either one transistor or the other is conducting. The triggered condition will remain, until the circuit is triggered (flipped), to the opposite condition, thus we have a simple memory.

Static memory was originally based on

'bi-polar transistors'. Again, I won't bore you with what they are here, suffice it to say that the original versions consumed vast quantities of power, and became quite hot to touch. This presented the major limiting factor with this type of memory, because if the transistors became too hot, they failed. To keep the temperature effects down, and because of the IC manufacturing technology available at the time, memory sizes were limited in size to two or three thousand bytes.

Dynamic memory was designed to overcome the limitations of bipolar static memory, but had to await the invention of CMOS (Complimentary Metal Oxide Silicon) transistor technology. In this type of memory, the memory element is a capacitor, which is charged to represent one binary value, or discharged to represent the opposite binary value.

As those of you with some technical knowledge will know, there is no such thing as a perfect component in electronics. The capacitor used as memory, suffers from this affliction, which causes the binary unit represented by a charged condition to fail, due to 'self discharge' of the capacitor (A bit like leaving a torch permanently switched on).

To compound the problem, the memory capacitor value is exceedingly small, unmanageably small in fact, which means the charge leaks away very rapidly. This is the major limitation with this type of memory, but you guessed it, the problem has been overcome, as we will see shortly. The major advantage of this type of memory, is its 'density'. Compared with static bipolar memory, a vast improvement in memory size is available, for the same size slice of

silicon. (As an aside, the use of CMOS technology in the manufacture of static memory, has also caused a considerable increase in memory density).

This improvement in density is due predominantly to the reduction in CMOS transistors needed for the memory to operate. The transistor this time has only to act as a switch, allowing the capacitor to rapidly charge or discharge on demand. It might be thought that bipolar transistors would achieve the same ends. Unfortunately, they will not, as they only compound the capacitor discharge problem.

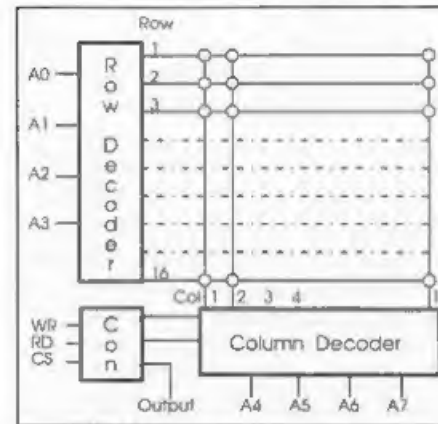
This self discharge problem is resolved by 'refreshing' the capacitors at regular intervals. Refresh is not as complicated as it may seem, the major requirement being that it is done at about 2m/Sec intervals, (2 one thousandths of a second). This sounds a very short space of time, but when compared with microprocessor operating speed, hundreds, even thousands of bytes of code can have been processed in the same space of time.

The way in which refresh is carried out is dependant on the microprocessor in use. The Z80 for example has a register, and a control pin dedicated to just this problem. For other microprocessors, a separate chip called a refresh controller could be used, or it may be done using a DMA controller (Direct Memory Access), or even by the memory chip itself.

I will return to this subject later. Before I do, we will need to look at memory construction.

The memory cells, each of which stores just one binary digit, (bit), are invariably arranged in matrix form, and generally a square matrix, ie. an equal number of 'rows' and 'columns' of wires connecting the cells, is used.

Typically, this might look like this:-



note! data lines omitted for clarity.

The above figure shows a few memory cells in a typical matrix configuration. For the above example of 4 binary row addresses, and 4 binary column addresses, access to any one of $16 \times 16 = 256$ individual memory cells is provided. The decoders take the binary input from the Address bus and each produces one (decimal) output, corresponding to the input required to select one row and one column. Each cell is enabled by its row input, so the chosen column will either accept a 0 or a 1 from the columns data bus in the WRITE mode, or put on the data bus that column's state of 0 or 1 in the READ mode. The WRITE and READ property is controlled by additional logic to the column drivers.

Each chip may be organised in one of 2 ways. First, the cells in any one ROW may form complete bytes, ie. from the above example, a chip organised this way would hold 32 bytes of 8 bits each. The second type of organisation allows only ONE bit to be selected. Again from the above example, a 256×1 bit memory is produced.

It follows that with 8 row and 8 column

address lines, 64K cells are addressable. These can be organised as $64K \times 1$ bit, or $8K \times 8$ bits or $16K \times 4$ bits etc.

The memory size becomes the limiting factor now, because of the number of pins available on the outside of the memory chip. For very high density chips it is common to have 'multiplexed addressing', which will halve the number of address pins. In the above example, the four row addresses are first applied, followed by the four column addresses - to the same pins. It is normal to have one or two additional pins on the memory chips, to indicate a row or column selection, ie RAS and CAS.

The final requirement is some means of enabling the data output. As the data bus is shared by all memory chips, the CPU, I/O chips etc, the memory chip - along with the rest - is equipped with 'tri-state' I/O lines. These, as the name implies, have three states: high, low and disconnected, thus preventing loading of the data bus, and false data presentation, when two or more devices are operating.

To read or write data to the memory therefore, the following sequence is necessary:-

1. Set up the address (either all of it, or Row/Column).
2. Decide if a READ or WRITE is needed, (usually the same pin on the chip).
3. Enable the memory chip data line(s). This is normally done using a pin marked CS for chip select, or CE for chip enable.

Simple isn't it!!

In the simplified explanation above, I have omitted to describe how the data gets to the memory cell. The static type of memory cell is a little more complex as it requires two 'triggering' lines in order to set or reset the flip-flop, and hence the

value held in memory. It follows that our single data line from the computer bus must be treated in some manner to enable this to occur.

In practice the solution is very simple, the inclusion of an inverter. For those of you not familiar with this device, the output of an inverter is the opposite of the input. As an example, if our two transistor flip-flop is in a stable state, with transistor 1 off and transistor 2 conducting, applying a low level input to transistor 1 will cause no change. As transistor 2 input is connected to the output of the inverter, the input of which is connected to transistor 1 input, the level applied to transistor 2 will also cause no change.

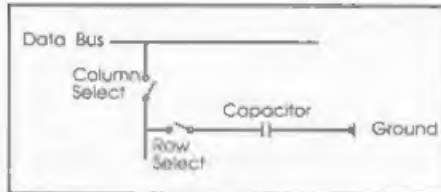
If however a signal of the opposite polarity is applied to transistor 1 input, it will cause it to switch on. The output from the inverter will also change, and this time will switch transistor 2 off. When we write to the memory cell it is left holding a value equal to the controlling input.

To do a read of the cell, it is only necessary to 'gate' one of the transistor outputs onto the data bus using the read control line.

All this makes it sound as if a lot of transistors are used for purposes other than as memory cells. IC manufacturers however have one or two tricks up their sleeves, and using the appropriate technology can use the two transistors to do all the necessary actions.

Dynamic memory, as discussed earlier, relies on the electrical charge held by an exceedingly small capacitor, and this charge must be regularly renewed in order to maintain the memory value.

This time we will represent the row and column selections as switches so:-



To access the capacitor first the row switch is closed, (by the row decoder), connecting the capacitor to the column line. Next the column switch is closed, (by the column decoder), connecting the capacitor to the data bus. The capacitor may now be read from or written to, in the normal manner. Simple? The problem of course occurs when we try to do a refresh.

In reality it is very simple. Lets take the Z80 as an example again, it is after all used in both Speccy and SAM and of the 8 bit micros is, I believe, the only one to contain on board refresh control. As mentioned earlier the Z80 contains a dedicated piece of memory called the Refresh register. This 7 bit register contains the processor incremented address, of the dynamic memory locations currently being refreshed. (The full memory address for large memory is usually completed using the Index register).

The 7 bit address is placed on the lower 7 bits of the address bus, and is used to control up to 128 rows via the memories row decoder. The Z80 also outputs a control signal, RFSH, which is connected to the memory chip RAS (Row Address Strobe) pin. MREQ is also used to enable the memory, but RD and WR are held inactive. These actions normally take place during an M1 cycle T state. I won't go into what this means here, except to say that the micro-processor is normally 'inactive' at this time.

So all we need to do for a refresh is just

toggle the RAS line. This is of course done every time a memory access is made for reading and writing, but it will be very improbable that every memory cell will be used regularly with every program.

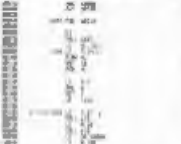



The actual refresh of dynamic memory cells is usually done by the memory itself and normally as a complete row of cells. The theory behind the refresh is very simple - *in theory* - but very complicated in practice.

Theoretically, all that is done is to compare the actual level appearing on the memory cell, with a level determined by the IC manufacturer. If the level on the memory cell is found to be greater than the reference level, then the level in that cell must be a charged condition. The 'sense amplifier' which performs this task, then recharges the cell. If the

tested level is lower than the reference level, it's taken to be an uncharged condition, and no further action is taken.

The practice behind this simple principle, is very complex, and took many years to resolve. The addition of more circuitry on the silicon chip, (sense amplifiers), caused a small reduction in memory size, and an increase in power requirements, as the sense amplifier uses flip-flops. Fortunately for us, it is not a severe problem, and we now have high density memory, with reasonable power requirements, (all the 'portable' PC's now available bear testimony to this), at a reasonable cost.

I hope you have found this brief insight into RAM useful, and that it will encourage you to investigate a little further.

<p>If you are new to the fascinating world of machinecode, SC_ASSEMBLER is the ideal program to help you, all thanks to the unique automatic autotab editor designed to help the absolute beginner.</p>	 <p>SC_ASSEMBLER £12.50</p>	<p>SC_MONITOR is an essential utility to use with an Assembler to help understand and debug machinecode programs. Single step each opcode to see what effect it has on flags, registers, memory and the screen. TurboMON another good monitor program is also included free, as well as a database on opcodes.</p>	 <p>SC_MONITOR £15.50 (£24 with Assembler)</p>
<p>The most powerful word-processor for the Sam Coupe, the only one to output built in hi-res fonts, screen pictures and drawing lines with the same quality as more expensive software for Amiga and PC Computers. Can load in and print your own word-processor files for the Secretary, SpellMaster, Tasword etc. 25 page A4 professional comb bound manual.</p>	 <p>SC_WORD pro £29.99</p> <p>SC_WORD pro INTRO £14.99 (without hi-res font).</p> <p>Need MasterDos Any Printer except 300dpi Printers and Lasers.</p>	<p>SC_FILER the only powerful database programmed in 99% machinecode. You can set up the database to your exact requirements, with fields of any size, position, colour and charset in 4 different Mode 3 layouts and printouts.</p>	 <p>SC_FILER £12.50</p>
<p>STEVES SOFTWARE 7 NARROW CLOSE, HISTON CAMBRIDGE, CB4 4XX. TEL 01223 235150 after 6pm</p>			

**SAMDISK Issue 16 £3.00**

The best Samdisk ever! 2000k for just £3! Includes Freddy Frog Arcade playdemo, SAM Coupe Encyclopaedia playdemo, Shrapnel 2 complete game, Samdisk Movie Demo, SAM Paint Guide Part 2, SAM Paint customiser, SAM Paint exclusive fonts, GamesMaster guide, Football League Manager Superhacker, Amalthea 2 and Grubbing for Gold screenshots - the first EVER, Crossword, Rag's Hacker, Manga Reviews - over 20 videos analysed, Marbles Deluxe Designer 2, Lee Willis's Filer '95 filing system, plus tons more!

MARBLES DELUXE £6.00

Arcade-puzzler with 50 levels, 512k animations, built-in screen designer, passcode system, 512k music by Lee "Bubel" Willis, 256k version on the same disk, plus more! Coded by Steven "Pickasso" Pick.

PICKASSO III £3.50

The largest and greatest collection of ORIGINAL SAM works of art from the SAM's graphics maestro, Steven "Pickasso" Pick! Super-compressed and WIMP driven, almost 100 screens can be loaded, admired and read about - and they're all ORIGINAL!!

ALSO FROM ATOMIK...

SAMDISK 1-4 £1.50 each **SAMDISK 5-15** £2 each **SAMDISK OMNIBUS 1-3** £2 each **PICKASSO PORTFOLIO 1** £2 **PICKASSO PORTFOLIO 2** £3.50 (2 disks) **BAKTRAK** £2 **ALL SOFTWARE WORKS ON SAM ROM 3.1 AND ABOVE.**

ORIGINAL 2nd HAND SPECTRUM SOFTWARE (send SAE for full list for titles for sale). **COMING SOON - SAM ENCYCLOPAEDIA 1995** (Sept '95), **SAMDISK 17** (Dec '95), **FREDDY FROG** (Jan '95 - honest!).

CHEQUES and PO's TO STEVEN PICK**ATOMIK SOFTWARE Dept F, 20 GROVE ROAD,
HOYLAK, WIRRAL, MERSEYSIDE, L47 2DT.**

HOW BASIC IS STORED

By:- Chris Lord.

Have you ever wondered just how your Spectrum knows what to do when you switch on? Spectrum Basic is already there, awaiting your command but how does it understand what you want? Well, the general principles are what I want to look at in this article. Other micros, especially SAMs understand, are pretty much the same so there will probably be something in this article for you even if you don't own a Spectrum.

Perhaps the most important difference between Basic and other high-level computing languages such as Fortran or C is that Basic is an interpreted language, whereas the latter two are compiled. What this means is that while a program written in Fortran or C is converted into machine code and then loaded and executed, one written in Basic is loaded and then read statement by statement, each statement being interpreted and acted upon by the machine code in the ROM which is designed to handle all legal variations of each type of statement. There are compiled Basics of course, but they are outside the scope of this article.

This 'interpretation' has various effects, both good and bad. The ones that most concern us here are that it makes Basic rather a slow-running language - although this isn't too apparent on most modern machines unless you go in for a lot of screen animation - and it means that the source program (the Basic program that you actually wrote)

remains in the computer's memory all the time it is being executed. And that in turn means that it is very easy for us to watch it working.

Now it's time to actually poke around in your Spectrum's innards - or PEEK around, to be more precise. Because there are no hard and fast rules about where Basic programs are stored, the first job is to discover the beginning of the program storage area on your set-up.

This information is contained at locations 23635 and 23636, and you can discover it by keying in the following as a direct command:-

```
PRINT PEEK 23635 + 256 * PEEK 23636
```

We'll see the reasoning behind this later.

The information is actually contained in a system variable called **PROG**. If you don't have microdrives or strange disc systems then you will most probably get the answer 23755, and this is the number that we shall use in the examples that follow. But if the answer you got was different, use the number that you got instead.

Now type in the following Basic program exactly as it is here:-

```
10 REM This is a remark
20 PRINT "This is a PRINT statement"
```

Then key in the following direct command:-

```
LET START=23755;FOR I=START TO START+21:PRINT PEEK I;" ";NEXT I
```

and the computer should respond with this sequence of numbers -

```
0 10 18 0 234 84 104 105 115 32
105 115 32 97 32 114 101 109 97
114 107 13
```

What you are seeing here is how the first line of the Basic program you typed in is stored in the computer. Each number represents the contents of one byte of memory, starting from the point at which your program begins. As you probably know, a byte can store any number in the range 0 to 255.

The first two numbers, 0 and 10, are the number of the first line of the program. It just so happens that the Spectrum stores line numbers in the same order that humans regard as correct, with the high number first and the low number second.

For hardware reasons, however, computers store many numbers in the 'wrong' order, with the low number first and the high number second. To work out the number encoded in the next two bytes, you multiply the high byte by 256 and add it to the low byte (just as we did earlier when we PEEKed the system variable PROG).

The next two numbers, 18 and 0, mean that this line of program needs 18 bytes to hold it - not including the two bytes used for this information, or the two that store the line number.

We arrive at 18 by multiplying the 0 - the high byte this time - by 256 and adding 18, our low byte. Got it?

After this comes the code 234 which stands for the Basic keyword REM. All the standard Basic keywords - ZX Basic's vocabulary - are represented in a similar way, and the codes are more often called tokens. The simple rule here is that every command which you can input on

the Spectrum by means of a single keypress is represented by a token - even in 128K Basic, where keywords are typed in character by character, the editor still converts the keyword into a token before storing it in memory.

Note that the spaces which the Spectrum automatically inserted after the line number and again after the keyword REM are not represented in the memory. They only appeared to make it easier for you to read the line. (Most tokens have at least one extra space included in their printed form but putting some tokens together can produce two spaces, try it and see.)

Anyway, now it gets simple. If you look up the next 16 numbers (as far as the 107) in Appendix A of the Basic programming book, you will see that they are the character codes for the remaining letters in the statement (32 is a space, 84 is a capital T and so on). Finally, the sequence finishes with 13, which means Enter, as an end of line marker.

All Basic command statements are represented in the same general way. If you're still confused, Fig 1 (on the next page) may help to make things a little clearer.

By now you have probably worked out how to inspect the representation in memory of the second line of the program. Clear the screen (to keep things tidy) then enter the following direct command -

```
LET J=23777: FOR I=J TO J+32: PR
INT PEEK I; " ";NEXT I
```

and the computer will respond with -

```
0 20 29 0 245 34 84 104 105 115
32 105 115 32 97 32 80 82 73 78
84 32 115 116 97 116 101 109 101
110 116 34 13
```

You know now that the first two numbers show that this statement is line number 20, and the two numbers after those (10 byte/h1 byte - remember?) inform you that it is 29 bytes long. Then comes the token 245, which stands for PRINT, a double quote - 34 - and so on.

But note that here the word PRINT, which formed part of the message included in the quotation marks, is listed as a series of character codes (80,82,73,78,84) rather than as a token, because tokens only stand for instructions that the computer has to obey and not for similar words which occur inside quotation marks.

And finally there is the Enter token - 13 - to confirm that we have come to the end of the line.

At this point you would probably assume that all Basic statements are represented in memory in the same way. Actually this is not quite true.

Get rid of your program by entering NEW, then enter the following:

```
10 LET p=1
```

then re-type a version of our familiar direct command line to see how that program was represented.

```
LET J=23755: FOR I=J TO J+14: PR
INT PEEK I; " ";NEXT I
```

You should get the response:-

```
0 10 11 0 241 112 61 49 14 0 0 1
0 0 13
```

Some of that is certainly familiar enough. You will recognise the line number - 0 10, - the line length, - 11 0 - and the Enter at the end - 13.

Other codes which shouldn't cause any problems are the token for LET - 241 - and 112, 61 and 49, which are the character codes for the variable name p, the equals sign and the number 1 respectively.

The rest of the line is a little more difficult. Try entering different values for the variable 'p' in the program (both integers and floating point numbers) and see if you can work out for yourself what is happening.

You want a clue?

Well OK, just this once then.

In statements that include numbers, Spectrum Basic represents the numbers not once but twice -

- once for the purposes of LISTing and
- once for its own internal operations.

That should point you in the right direction and give you lots to think about until we meet again.

I will fill you in on what is really happening next month.

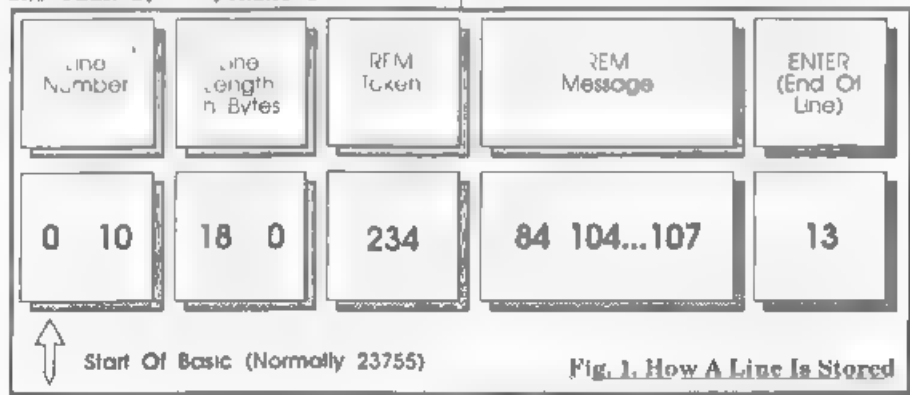


Fig. 1. How A Line Is Stored

SAM GAMES AVAILABLE FROM REVELATION

SOPHISTRY

Our new licenced game. Originally produced for the Spectrum by CRL, and now brilliantly converted for SAM, Sophistry is a game that is big, perplexing, colourful, frustrating, musical, bouncy, and above all DIFFERENT (and it even has the Spectrum version built in so you can take a trip down memory lane)

£9-95 (£8-95 to INDUG members)

NEW TWT NEW

A game packed with humour, colour, sound and above all ACTION!

Loosely based on one of the best Spectrum games of all time Bomb Jack (over the years the game SAM owners have most requested) As well as the normal one player game, we have gone one better with TWT by giving you two player action - yes TWO players on screen, competing against each other, in full glorious technicolour - at the same time. It even has a novel feature in a SAM game - a High Score Table, how many others have that?

Avoid the meanes, collect the bonus points, Jump, Hover, Fly - what more do you want???

£9-95 (£8-95 to INDUG members)

PRINCE OF PERSIA

The classic game of sword-fighting and adventure that has you racing to save the beautiful princess from her fate at the hands of the evil Vizier. Superb animation, hundreds of rooms to explore, a real challenge for any games player. Original priced at £14.95 now re-released by popular demand and at a price that no-one can miss.

£9-95 (£8-95 to INDUG members)

LEGEND OF ESHAN

A game like none seen before on SAM.

A graphic game of strategy and adventure taking you into a vast world of demons and war-lords. This is *not* an arcade game, this is *not* an adventure, this is something very special.

Legend Of Eshan works with keyboard or joy-stick and is also fully Mouse compatible. Comes complete with comprehensive instructions and a map of the lands of Avnall.

£14.95 (£12.95 to INDUG members)

More Games Coming Soon

Send SAE for full list of available games

REVELATION FOR THE VERY BEST IN SERIOUS SOFTWARE

► **THE SECRETARY** - The most advanced word-processor for SAM, powerful, versatile and easy to use. **NEW Version 1.5 Available Now** - The BEST just got better £14.95 (INDUG £12.95)

► **COMET Z80 ASSEMBLER** - The SAM assembler many professionals use, yet easy enough for the beginner - so why settle for second best? £29.95 (INDUG members £21.95)

DRIVER

The Graphical User Interface for SAM

Comes complete with many built-in utilities and ready to run applications. Full manual and a disc-based tutorial DRIVER gives you the ease of use only found on big office computers before. So easy to use yet so powerful. **Fantastic Value £29.95 (INDUG members £24.95)**

Special Offer - Buy DRIVER together with an Official West Coast Computer's SAM Mouse and **SAVE £5**. Our price ONLY £69.95 incl UK p&p (INDUG members £64.95)

NEW - SCADs PD Yes, at last we are please to re-release SCADs - the arcade game development system for SAM. Previously sold by Glenco at £24.95 we have now placed the software into the Public Domain so the disc costs you **Only £2.50**. The full manual (over 200 pages) is also available for £12.95 (overseas please add an extra £1 to postage rates below)

- All prices include UK postage and packing (Europe please add £1 other overseas please add £2)
- Payment by UK cheque (made payable to Revelation Software), PO or cash. Sorry but we can't accept orders by Euro-cheque or Giro

Please Quote INDUG membership number clearly on order if claiming discount.

Send orders to

**Revelation
Software**

Revelation Software, 45, Buddle Lane, Exeter, EX4 1JS.

EXPLORING SAM MIDI

By:- P.J.Williamson.

MIDI is short for Musical Instrument Digital Interface. It is a system that enables one or more electronic keyboards, drum sound generators etc to be controlled from a central computer.

The MIDI system is a serial data system similar to the well known RS232 and RS423 systems except that it uses a baud rate of 31.25Kb. Without going too much into the technicalities of the system, the signals that go from the computer to the keyboard start with a byte of 10 binary bits, 1 start bit, eight data bits and 1 stop bit. In decimal notation this signal is equivalent to 143 plus the MIDI channel number being used. This tells the keyboard that a note is about to be switched on. The next signal sent uses the same 10 binary bits to send the note reference number. This can be from 1 to 127 in decimal (one for every semitone, so over 10 octaves can be catered for. Middle C is reference number 60). The final byte sent is a number from 0 to 127 decimal. 0 is no sound and 127 is maximum volume.

On keyboards the sound volume is not often adjustable as the keys are not velocity sensitive, so the full volume signal is sent. On the writer's keyboard any value from 1 to 127 gives the full volume. On a piano by contrast this signal would control the volume of sound individually for every note pitch sent.

The above sequence of signals switches a given note on. It will now remain on until it is switched off either by a MIDI

signal or by cutting off the power to the keyboard.

The MIDI 'note off' sequence starts with a warning byte, in this case 127 plus the channel number. This tells the keyboard that a note is to be switched off. Next comes the byte identifying the note by its pitch number and finally a byte setting the note volume to zero.

The above sequence of events happens for every note switched on or off.

There are 16 channels available in MIDI for sending out note control data. A single keyboard may only use two or three. The others are available to control other electronic musical instruments such as drum, sound generators etc. On a keyboard there will also be channels for switching instrument sounds, either prior to the start of a piece of music or during the performance of the music. On my keyboard channel 49 controls the treble or melody instruments whilst channel 50 controls the accompaniment sounds.

The instruction manual for a given keyboard should give this information. If it does not then one has to feed test signals into the keyboard until it responds. After channel 49 or 50 is selected then one puts out a second byte, as if it was a particular note switch on signal, only in this case the signal switches on the instrument. Again the third byte the volume signal is sent, to keep the system happy, although it does not do anything.

Unlike the note sequences, one does not have to send a switch off sequence on changing an instrument, you just send another switch on sequence with a new instrument reference number

For the newcomer to MIDI, it is very interesting and instructive to generate MIDI instructions using Basic. However the response obtained with this language is too slow for a satisfactory sequencer program to be written. After initial attempts with a Basic program I went on to develop a sequencer using machine code. The program finally developed used Basic to load music data and machine code to feed control signals to the keyboard/piano.

The following two Basic programs will enable the newcomer to MIDI to produce sounds from a keyboard. Both programs are very similar. Both send control and data signals to SAM's MIDI port, 253 decimal.

```

10 REM MIDI KEYBOARD TESTER
20 PRINT "ENTER CHANNEL NUMBER"
30 INPUT C: CLS
40 PRINT "ENTER INSTRUMENT NUMBER"
50 INPUT I: CLS
60 OUT 253,143+C: OUT 253,I:
  OUT 253,127
80 PRINT "ENTER CHANNEL NUMBER"
  R""FOR PLAYING CHANNEL"
90 INPUT C: CLS : LET VOL=127
100 DO
110 READ PTCH,DUR
120 IF PTCH=0 THEN STOP
130 OUT 253,143+C: OUT 253,PTC
  H: OUT 253,VOL
140 FOR X=1 TO DUR: NEXT X
150 OUT 253,143+C: OUT 253,PTC
  H: OUT 253,0
160 LOOP
170 DATA 60,200,62,100,64,200,
  65,100,67,200,69,100,71,10
  0,72,400,0,0
180 STOP

10 REM MIDI PIANO TESTER
20 PRINT "ENTER CHANNEL NUMBER"

```

```

R""FOR PLAYING CHANNEL"
30 INPUT C: CLS
40 DO
50 READ PTCH,VOL,DUR
60 IF PTCH=0 THEN STOP
70 OUT 253,143+C: OUT 253,PTC
  H: OUT 253,VOL
80 FOR X=1 TO DUR: NEXT X
90 OUT 253,143+C: OUT 253,PTC
  H: OUT 253,0
100 LOOP
110 DATA 60,20,200,62,30,100,6
  4,40,200,65,50,100
120 DATA 67,60,200,69,70,100,7
  2,120,400,0,0,0
130 STOP

```

The Keyboard version allows the switching on of instruments but with fixed sound volume, whilst the Piano version allows the individual note volume to be controlled.

Using the Keyboard program, the keyboard will revert to instrument number 0 when it is switched on. When the correct channel is entered on the test program and the instrument reference number is selected, the indicator light on the keyboard will jump to the instrument selected. If the correct music channel number is now entered, the test program will play a scale, with the instrument sound previously selected.

Readers I am sure will be able to develop more sophisticated programs on these lines. However those given should enable you to find out which channels do what on your particular instruments.

During the development of the programs I have written, having only a keyboard on which to test things out I was lucky to obtain the help of another *FORMAT* reader who had an electronic piano. He was able to test the piano version of the program.

An interesting byproduct of this co-operation was that another version of the programs developed. My method of music data input was produced for a

person that could read music and could enter the note names and octave numbers. My friend produced a data input method that used the MIDI code numbers for the notes. This meant referring to a suitable chart or list, but nevertheless the method of entry was quick and very effective. All four versions should be available.

I have sent the programs described below (and several other programs that I have developed over the years) to the SAM Public Domain Library. Those readers that are interested in trying out these programs, should contact the library by writing to

Derek Morgan,
Sam Public Domain Library,
18, Mill Lane,
Glenburn Road,
Old Skelmersdale,
Lancashire.
WN8 8RH.

Enclose a stamped addressed envelope and he will tell you how to send for the software on disc.

Now for some detail about the programs. The machine code program is placed in RAM from address 33000 to 33499. Music note data is put into 33500 to 49499 whilst channel data goes into 49500 to 65499 in the case of the keyboard programs and note volume for the piano version of the program. These two blocks of data are scanned in tandem at a steady rate. Imagine going from each address and peering at what is there at a steady rate. Each address can be regarded as having a unique time point in the overall time it takes to cover the 16K of RAM. When a note is required to be switched on then its MIDI pitch number is put at the RAM address that corresponds to that particular time point. If the note is to be switched off then the

MIDI pitch number plus 127 is put into the appropriate RAM address. At the same time the channel number for keyboards or the loudness factor goes into a matching position in the second RAM section from 49500 onwards. This data is poked in via the Basic program.

When in the play mode, the machine code program looks at each RAM address. Where there is a 0 the program goes through a dummy routine so that the time taken to sample each address takes the same time whether data is there or not.

The SAM is used in the unexpanded RAM mode. This means the RAM left for the Basic program is very tight indeed. I would have liked to have used the paging system so that the whole of the SAM's 512K could have been used. Why didn't I do this? Simply because I did not know how to do it so if there are readers with this expertise perhaps you could enlighten me as to how this is done. I don't think *FORMAT* has ever covered this in detail, has it?

As the MIDI system is a serial data system, only one piece of data can be transmitted and received at any one time. Where simple music of one note at a time is being used then there is no problem but where the music comprises multi-note chords or different instruments on different channels are being controlled then the problem arises that notes which should start or stop at the same point in time cannot do so. The best that can be done is to switch the notes one after another in a very short space of time so that to the listener they appear to all start or stop at the same time.

A computer program using Basic cannot do this and the chords now sound as arpeggios, the chords sound as

individual notes starting after each other. Using a machine code program speeds things up about 300 times according to my measurements. Instead of one time point being allocated to the shortest note in the music and longer notes taking up more in direct proportion to their duration, we have to allocate say 100 time points per single beat of the music. In this case a chord of three notes would have a spread of 3 hundredths of a beat on start or stop. This gives quite acceptable results. However, if we use 100 time points per beat, the total length of music we can store is 16000/100 or 160 beats of music. With music in 4/4 time this means a length of 40 bars of music.

The program allows the user to select the number of time points per bar for a particular piece of music. One has to select a figure that gives acceptable chord rendering whilst not using up too much RAM.

The tempo of the music on replay is controllable over a wide range. After each sampling of a byte of data in RAM, a counter subroutine provides a delay before the next step in the data sampling takes place. A target number for the counter is poked in from the Basic program. A small number, say 10, gives an impossibly fast rendering of the music. Normally values from 300 to 600 are about right. Values over 600 give funeral renderings where each individual note in a chord can be heard. This can however be very useful when checking the accuracy of your data entry.

Turning now to the Basic part of the program, after initially setting up the time points per beat, channel number required etc., the actual music data is entered, one note at a time. Note name, natural or sharp or flat, then octave and duration of the note in beats or decimals

of a beat, in my version of the program. The other version allows for the entry of two digits for the note MIDI number, then another two digit number for the note duration and then a 1 or a 2 indicating whether the note stands alone or is part of a chord. All these 6 digits are entered together making the entry quick and efficient. The former method of entry requires a separate entry of the 1 or 2 indicating single note or part of a chord. However the former method allows great flexibility in the duration of a note to be set in, useful for experimenting with exotic note timings. (See my later description of my 'Duration' program).

A menu option allows for the scanning of the data in RAM giving a display of data against time point. This is invaluable when changes of the entered data are required. An option allows for the deletion of data at a given time point, another allows for deletion of a whole block of data between set time points. These routines run in machine code and are very fast in operation.

Another option allows for the repetition of a given phrase of music between set in time points. Any number of repetitions can be provided. A further routine allows for the movement of blocks of data forward or backward in time. As these last two routines run in Basic they do take considerable time where large blocks of data are being handled.

When a piece of music has been entered into RAM it can be stored on disc for future use or further development. In addition to the music data all the relevant program variables are stored.

The data stored for the Piano version of the program is not viable for use on the Keyboard version and vice versa

This is because different data has to be stored for the two programs. Data is however viable between the two different versions of the piano program as the only difference in these programs is the way the note data is entered. This applies to the two versions of the Keyboard program.

Another entirely separate program is also contained on the program disc. This program is called 'Duration' and is very useful for developing your own timing of the music. When a piece of music is written down, the note durations shown are really a guide to the musician as to the approximate length each note should last. The musician puts his own stamp on the musical rendering by subtly lengthening or shortening notes to give character to the music.

If music is played with the exact timings as written down (and this is what can be achieved with a computer driven performance) the music can sound very mechanical. To use the 'Duration' program one plays the tune rhythm on one key whilst humming the tune or thinking the tune through in one's mind. The end of the sequence is made by pressing the space bar.

The relative timings of each key press are then displayed, one display showing the time the key is being held down and the time when it is not, whilst another display shows the time between the key being pressed down to the next time it is pressed down.

The data is now processed so that the timings are relative to a given number of bars of music. The timings listed can then be used in the data entered into the previous piano and keyboard programs to give the subtle timings that bring music alive.

I hope this article will be of interest to

readers and I hope those that have an existing or awakening interest in MIDI can get hold of an electronic keyboard or piano and try out firstly the simple programs in Basic to get a feel of their instrument's possibilities, and then trying out my programs. I should mention that I regard these programs as very experimental and capable of further improvement and development by those readers with the expertise in machine code programming. The programs would be much improved if the whole of SAM's expanded RAM could be used for the storage of music data. A method of directly entering the music onto a displayed musical staff would be another great improvement. The one advantage of my programs is that multi-note chords can be handled. You are not tied down to single notes per channel.

Now I don't know whether the book is still available, but I would suggest that anyone that wishes to study the MIDI system in more detail and maybe make up their own MIDI interface unit for say a Spectrum, should obtain a copy of MIDI Projects by R.A. Penfold and published by Bernard Babani. The book reference number is BP182.



VARIABLES ON A THEME

Part 3.

By:- Dilwyn Jones.

OK, ready for another dose of information on the Spectrum's system variables? Good, cos you are going to get it anyway, so sit back and take your medicine

If you remember from the first two articles in this series, system variables are those bytes in memory that help the Spectrum to remember certain things it needs to know about itself, they are the storage locations used by the ROM routines that make the Spectrum work

I will continue with them in order

- 23615/8
- LISTSP

LISTSP is an internal address used by the ROM to keep track of where it was when it stopped to do a listing on the screen

- 23617
- MODE
- Specifies cursor.

Values zero, one, two or four specify the L/C mode, E mode, G mode or K mode respectively. POKEing this system variable will affect the appearance of the cursor - it may appear as a flashing letter, number, symbol or even a key word.

This is most apparent during an INPUT statement. The value is read when the need arises, e.g. a mode change made normally from the keyboard. So if you get into difficulties, press both Shift Keys for E mode and then the same again to get back to normal L/C mode.

Try this program which POKEs all possible values into 23617. Most are variants on the four cursors, ie. you will find yourself in a particular mode after the POKE, such as everything coming as graphics as in G mode. 252 will give an L/C mode flashing '<' to point to where you're typing.....

```
10 FOR A=0 TO 255
20 PRINT A
30 POKE 23617,A
40 INPUT AS
50 NEXT A
```

Some values are obviously more useful to you than others, so make a note of any you think you may use in the future

- 23618/9
- NEWPPC
- and
- 23620 NSPPC

23618/9 is a two-byte system variable containing the line number of the line to be jumped to. 23618 contains the lower byte of the line number contained is read as -

```
PEEK 23618 + 256*PEEK 23619
To POKE a line number in, say line X.-
POKE 23618,X-256*INT (X/256)
POKE 23619,INT (X/256)
```

We now come to system variable 23620. With 23618/9 and 23620, we could actually simulate a GOTO command to a statement within a program line, should that ever be necessary. GOTOs cannot access individual statements within long program lines.

To jump to statement four in line X, first go through the motions described

above then POKE 23620,4 and the jump is executed

- 23624
- BORDCR

The bits of this system variable control the attributes of the lower screen and the BORDER colour. Take a look at this table -

BIT	USE	VALUE
7	FLASH	0 or 1
6	BRIGHT	0 or 1
5) BORDER Colour	0 to 7
4) and PAPER for	
3) Lower Screen	
2) INK Colour	0 to 7
1) for	
0) Lower Screen	

By POKEing various values into this system variable you could achieve a flashing, bright, multicoloured lower screen, or make both PAPER and INK the same colour to prevent other people getting at your programs - however, any alteration would have to be made blind. You could also make INPUTs extra bright to stand out.

- 23627/8
- VARS

The pointer to the start of the variables store. Apart from finding your way into the variables area, you can find the length of the Basic program with this expression -

```
LET bytes=PEEK 23627+256*PEEK 23628-PEEK 23635-256*PEEK 23636
```

This excludes the screen, system variables, stacks and variables

- 23629/30
- DEST

The address of the variable when it is assigned to. If the variable had been set

up before, it would point to the start of where it was stored in the variables area. If it was being defined for the first time, it would point to the address of the start of the name of the variable in the program, eg. in 10 LET A=5, it would point to the address of the letter A.

It can also be used to find the memory address of a numeric variable, if you use something like LET A=A as in the following program:-

```
10 LET A=5
20 LET A=A
30 PRINT PEEK 23629+256*PEEK 23630
```

- 23631/2
- CHANS

Stores the address of where the channel information area starts. Streams and Channels deserve whole articles to themselves so I won't go into detail here

- 23633/4
- CHURCHL

The address of INPUT/OUTPUT information used at that moment. It normally points during an INPUT/OUTPUT operation to a five byte block of data in the channel information area. Use this program to examine the contents -

```
1 FOR X=0 TO 3: PRINT #X; PEEK
K 23633+256*PEEK 23634: NEX
T X: PAUSE 0:STOP
```

- 23635/6
- PROG

The address of the start of the area in memory where the Basic program is stored. This points to the first byte of the line number of the first program line. This may be useful if you're converting programs for other computers with information held as a REM statement in the first line of a program. See also

under VARS above.

If you wish to 'security-lock' a line into a program, then by means of this system variable you could POKE a zero into both bytes of a line number at the start of a program. Program lines start with a two-byte line number

- 23637/8
- NXTLIN

The address of the start of the next program line. You could use this to enable you to access machine code stored within REM statements anywhere in the program, eg. those loaded with MERGE from a tape library of subroutines. These would have their own local calls to machine code like this -

```
9000 LET A=USR (PEEK 23637 +256*  
PEEK 23638 +5)  
9010 REM <> MACHINE CODE <>  
9020 RETURN
```

One constraint to this is that you should not include any colour, flash brightness, etc, control characters into the REM statement or they may be interpreted as machine code, upsetting things somewhat. However, if used from a library of subroutines, these would not normally be used anyway

- 23639/40
- DATADD

This contains the address of the comma ending the last item of data. If nothing was read from the list (eg. after RUN or restored, etc) the address held in 23639/40 is the address of the byte before the program area, normally the CHR\$ 128 at the end of the channel information area. To demonstrate try RUNNING this program -

```
10 DATA "1", "2", "3", "4", "5"  
20 LET A=PEEK 23639+256*PEEK  
23640  
30 PRINT A; TAB 9;PEEK A; TAB  
18;CHR$PEEK A AND PEEK A>31
```

```
40 READ B$  
50 GOTO 20  
and you should get something like this.-
```

```
23754 128  
23763 44 ;  
23767 44 ;  
23771 44 ;  
23775 44 ;  
23779 13
```

The address in this two-byte system variable can point to the Enter character - the colon signifying the end of the line or statement containing the data - or the address of the terminator of the last item of data

Oh dear! Yet again, no more room for me to continue. Still, if you are all very good then I will return next month. Don't worry loyal Spectrum fans, there is plenty more still to come.

This article is extracted from Dilwyn's book, Delving Deeper into your Spectrum ROM - first published in the UK by Interface Publications. Although it is now out of print you should be able to obtain a copy through your local library.

OUTLET

The Disk Magazine!

+D +3 PC state which you want!!

(PC not converted but pure VGA/HD)

Software, articles, ideas, advice, letters

No unzipping or guesswork...
WE RUN IT FOR YOU!

Share ideas, programs and problems with others. Clear, graphic presentation. A joy to explore new software and information!

Only £2.00 to newcomers!

Chevron Software, 34 Saltersgate Drive
Birstall, Leicester, LE4 3FF

(Producers of OUTLET monthly since 1987)



YOUR LETTERS

Dear Editor,

How to cure a rheumatic mouse

After 5 or 6 years intensive use, my mouse gave running troubles. It worked - but not smoothly any more. So I rang *FORMAT* to ask if they could deliver a new one. They could, but advised me to try and buy it in my own country (The Netherlands) first as it would probably be cheaper.

However, before doing so, I opened my mouse to see what the trouble was. The rubber coating on the guiding wheels were partly gone. Then I got a thought. With a sharp knife I stripped the rubber coating from all of the three wheels. I went to a bicycle store and bought a piece of tyre valve rubber. At home I cut with a very sharp knife two pieces of about 2mm off. Then with help of a needle I managed them over the bare guiding wheels. The pressing wheel was enclosed in a housing so I couldn't do a thing about that. I put everything together and surprise my mouse runs smoothly as if born again. I still have rubber enough for years!

Yours Sincerely, S.M.S.Kempees.

Thanks for the tip, although I would like to point out to readers that not all mice have rubber tyred wheels and it is often the case that you need to scrape deposits off the metal wheels. Ed

Dear Editor,

In a recent issue, you asked for suggestions on how to make *FORMAT* appealing to a larger readership. No doubt most letters on this subject will be asking for coverage of different

computers, but there is still a way to create more interest from the Spectrum scene.

I have to say that at the moment, there is not much for me in *FORMAT*. As a Spectrum owner who doesn't have a copy of Taaword to fix (a fairly common subject in *FORMAT*, it seems!) and doesn't have much time to type in programs just for fun, there are only a few articles left. I unfortunately missed 'Machine Code Without Tears', but the Z80 Subs series is looking good.

Now I'm afraid I'm going to have to own up. I'm one of those PD demo writers. The PD demo scene is one of the most active things on the Spectrum. Like it or not, it needs coverage in *FORMAT*. I'm sure that if you lost your image of opposition to the PD community, some people would be willing to contribute.

After all, you claim to cover 'every aspect of computing except games reviews', not 'every aspect of computing except what the editor doesn't like'. Just because you're not interested in PD demos doesn't mean your readers aren't.

Yours Sincerely, Matthew Westcott.

As I've said before Matthew, we can only print what gets sent in. However, having said that, the Spectrum gets very good coverage in *FORMAT* and that will continue. What articles would you like to see? Carol's M/C series is still available if you order the back issues, so you don't have to miss out on that.

As to PD, if we are sent PD programs by their authors, giving details of where they can be obtained, then we will cover them in the news pages and/or arrange a

review. What we are not interested in, nor are our readers judging from the letters we get, is demo software just written to give 'scrolly' messages and flash graphics. I will do everything I can to encourage new programming talent and to help in the development of software (both serious or games) that is why INDUG was originally formed. My question to you would be "Why waste time writing a mindless demo when you could use the same effort to write a utility program or a simple game which would earn you far more glory and, just possibly, some money as well?". **Ed.**

Dear Uncle Bob,

Way-hey! I've never written to **FORMAT** before, so I thought I'd better, just to show that I do read it.

FORMAT has some great articles, but please include the odd bit of humour. I'm guessing that you could re-claim lost readers by just making things a bit jollier. I know **FORMAT** is a serious magazine, but a couple of the articles get a bit, dare I say, dull. The other idea I had to increase the readership was by including signed photos of me (swell).

Can you tell us all what West Coast are actually doing now, or is this one of those questions which you edit out? Everybody is getting a bit concerned - surely they should send out another newsletter to reassure us.

Even though all the arguments in the letters page are really petty (couldn't give a monkeys chuff whether onto has a gap or not), they're good for a laugh (if you're immature, anyway). So, I'm going to start another. Ready?

The SAM didn't take off because it's white. (ho ho, that should get people going).

Yours Sincerely, Colin Anderton.

Dull!!! **FORMAT**? Surely not?

Anyway, West Coast decided not to do another mail-shot as they are so expensive, they rely on **FORMAT** and

FRED to get their message across.

And as to the colour of SAM - I have always wanted a transparent SAM so I can see all the nice bits and pieces inside. **Ed.**

Dear Editor,

Re your editorial (Vol.8 N°10) regarding the expansion of **FORMAT**. One idea that has been used elsewhere (with apparent success), but may be of some help to you. That is the **FORMAT** readership advertise on free notice-boards at work, in supermarkets, in 'Free Ads' papers (twice weekly in this area), etc, etc, Wording to be either by you, or left up to the individual.

Secondly, and probably more expensively, flysheets could be printed by you and two or three sent out with each edition of **FORMAT** and/or one page of **FORMAT** (back cover?) could be printed up and, if possible, photocopied and distributed (again on notice-boards, etc) by the individual.

Thirdly, could you run an 'associate membership' advert in say, *Micro Mart* or similar. Like those you ran in *Sinclair User* and *Your Sinclair* in their dying days.

Anyway, hope to have been some help, or at least provided a start for better ideas. Thank you, and all those behind the scenes, for producing an excellent magazine twelve times a year. All the best.

Yours Sincerely, Ross Brown.

Ads in free papers and on supermarket noticeboards are a proven way of contacting others. If readers contact users in their area this way then word of **FORMAT** will soon spread. Anyone who wants a few subscription forms to hand out only has to ring Jenny and she will send them out with the next **FORMAT**.

The Associate Membership drive we had was successful and led to many new full members so we may give it a try again through other magazines. **Ed.**

Dear Editor,

Firstly, let me congratulate everyone at **FORMAT** for hosting the Gloucester shows. I look forward to the next one.

I have done some reading before buying a SCART lead for my new Sony KV-M1401U television, which I read in the letters section of **FORMAT** Vol 8 No 7 costs £7.95. Unfortunately, you also mentioned that the only TVs that it had problems with were Sony TVs.

This is in contrast with the Help Page (Vol 7 No 12) in which Ray Bray states how his older model Sony TV has worked perfectly with a standard SCART lead, and in a review of the Sony KV-M1400 TV (like mine but without teletext) in **FRED** 40, Matt Round also had no problems.

My TV's SCART socket is described as being CENLEC standard in the manual, which sounds as though most televisions use this pin layout, so I hope that your comment about Sony TVs was incorrect.

I have decided to bite the bullet and buy a SCART lead.

Yours Sincerely, Andrew Watkins.

Scart is one of those 'standards' that many break and I'm just as confused as you. Still, by now you will have received your Scart lead so I hope it worked for you. **Ed.**

Dear Editor,

I hope Jenny's wrist is not sore anymore, from my last letter regarding my problem with *The Secretary*, about the phantom 'I'. You asked did I have the latest version, yes I do have V 1.5.

I still get the phantom page numbers. Also I'm wondering does anyone else have difficulty with the keyboard, as Vers 1.5 is supposed to have a better keyboard response and I find that when I come to the end of the line and continue to type at the same speed I lose characters, as the program word wraps to the next line. Is there no way a buffer could be incorporated.

Also when a file is saved using the TEXT ONLY option I seem to lose a varying amount of characters off the end of the file.

I am looking for fellow War-gamers that may have good tactical/role playing games, especially:- SOLAR FIRE, NATO ASSAULT and DRAGONIA from Astros Prod' (What ever happened to them?) So come on, dig them games out, and SAM ones too.

Yours Sincerely, Alan R. Tilley (Elfie).

Can anyone help Elfie with his problems? We will pass on any letters as usual. **Ed.**

Dear Editor,

Today I received my first copy of **FORMAT** magazine and my membership letter. On the whole I must say I'm impressed with the content and presentation but I do have a couple of minor moans (Oh No!)

Firstly, The vouchers for the readers services page would be very useful to me, mainly the +3 DTP pack and the Music Maestro programs but only when finances allow. So as at the time of writing I only have 9 days to redeem them I shan't be able to take advantage of them.

Secondly, I own a Spectrum +3 with Multiface, the entire suite of Tasman programs, CP/M together with a Star LC/20 printer that enables me to do 'Serious Stuff' as well as playing games but it seems that without a PLUS D or DISCIPLE disc I am not able to use much of the software that is advertised in your mag. I am really thinking of the Ritman collection from B.G. Services, The Fractal Collection from SDS and Money Manager from you?

Is it not possible to supply these programs in a format for use on the +3 if its requested and the potential purchaser is prepared to pay the extra cost of a CF2 disc?

Perhaps I could appeal to these

The definitive database for the SAM Coupe ... is the *personal filing system*

from the keyboard of Jack Gibbons (author of the best Banking programs in the world!)

1000's of uses - can store anything - change it - sort - search extract - very easy to use - on screen help - scrolling summary displays with zoom, plus page up/page down, home/end, locate record - four print formats, summary, detail, label and mail merge - uses all available memory/disk space - massive data storage, over 1 MEG* - design/re-design screens any time - inbuilt totalling - field validation - table lookup/validation - WYSIWYG screen and report design - add/change your file structure any time - automatically restructures your data - uses index and random access files* - password control - operates on any SAM Coupe, 1/2 drives, 256/512K, can use 1 MEG memory expansion* - RAM drives - 50 page user manual - after sale support - auto-configures for 1/2 drives, 256/512K RAM, SAMDOS, MasterDOS/MasterBASIC, 1 MEG - no need to edit program

Send £19.95 (inclusive of p&p) specifying *SAMDOS(512K)* or *MasterDOS/MasterBASIC*

Warning ... not having a *personal banking system* can seriously damage your health!

If your finances are driving you mad, then we may have the answer to your problems. Over the last 13 years, the PBS has cured thousands of cases, resulting in very satisfied and sane clients.

Keep track of your finances, spanning all Bank, Building Society and Credit Card accounts - avoid expensive overdraft letters and bounced cheque charges - automatic posting of standing orders & direct debits for any frequency and for a set number of payments - extensive, flexible enquiries eg how much have you paid to the supermarket each month - forward projections to enable you to plan how much money you have left to live on - detailed Bank statements, more up-to-date than the ones from your Bank, etc - password controlled - on-screen limit monitoring so you know immediately if you are going into the red - Bank reconciliation so you can check the statement received from your Bank and see where they have gone wrong - suitable for private, club or small business - optional VAT facility - optional categories for analysis, account and GL codes with on-screen lookup feature - on screen help - 75 page user manual - after sale support

Prices (SAM Coupe/Elite - all models): PBS - £19.95; PBS Budget module - £9.95; PBS Final Accounts (POA). Versions are available for other computers eg BBC, CPC, QL, PC. PBS 1 for the Spectrum (all models) on cassette, microdrive, PlusD/Disciple and Opus for £12.

Hilton Computer Services Ltd. 3 Suffolk Drive, Guildford, Surrey, GU4 7FD

or phone 0483 578983 and pay on receipt of order. (*requires MasterDOS/MasterBASIC)

Buy with confidence - over 12 years in business: other products available - send SAE for details.

software suppliers and other readers to provide alternative versions or utilities to convert them to the standard +3 DOS.

I realize that it's up to us, myself included to send in articles or tips to enable you to give a decent coverage of the +3 and not just +2's or 128s with PLUS D drives and Sam Stuff.

I must just say that the mag is still a very enjoyable read aside from this and I will look forward to receiving it every month. By the way what date do you usually send them out?

In response to the letter from Mike Locker the CP/M manual from Locosoft does give some clues on extending the operating system to support a hard drive using FIDS (Field Installable Device Drivers) to hook in extra drives.

The manual supplied gives a clue to how to do this by looking at the RAMDISC.FID file supplied with CP/M.

On the subject of CP/M I would like to mention that I feel slightly conned at the claim that text files can be transferred using it from Amstrad CPC computers, PCW machines because although I can load these in they just throw up +3 DOS if I try to import my files into word-processors on these machines.

Perhaps someone knows of a way around this. Masterfile can do the job of converting text files but I can only get it to load in 32 characters across, useless for transferring Tasword files.

Also the vast majority of CP/M programs available are tailored for the PCW so if you are tempted to order a CP/M program you must check that it's Generic which means it will run on any CP/M machine.

There is a company called Advantage, based in Cheltenham, that do some language discs for COBOL and C. I fully intend to purchase these in the near future especially if Nev Young could be persuaded to do some follow up articles on C itself.

Again if anyone knows of a PD library

that has any CP/M programs that run under +3 CP/M perhaps they can let you know through the magazine or by writing to me at my home address.

I guess I have probably gone on a bit too long by now so I will sign off and say "FORMAT Magazine? Make it so Number 1". Bye for Now

PS (inevitable) I know how to protect +3 discs from Multiface 3 so it can't copy them. If anyone's interested let me know.

Yours Sincerely, Ray Smith.

The vouchers we send out to new members usually have at least a 3 months date limit, so you should have had plenty of time to use them. Ring Jenny if there has been a mistake on yours and she will sort things out for you.

Most of the software you mention will either not work on a +3 or are too big to go on a 3" disc. However, I'm sure that if enough +3 users started writing in then things may change - after all any business is led by demand. Perhaps if someone can write a few articles on the +3's DOS then more programmers will include +3 options in their software. *Ed.*

Dear Editor,

I could do with a bit of help please, I am not sure if this plea should go to one of the *FORMAT* experts, or go to the letters page, I leave the choice to you.

What it is I bought an AMX mouse, along with a second hand Spectrum set-up, the mouse being the part of it I was interested in. Anyway, it turned out to be a bit duff. One button would not work - this I traced back to one of the chips in the interface - changed the chip and it now works OK with 'The Artist II' and 'OCP Art Studio', but its printer interface will not drive my Citizen 120D printer, (I'm not overly concerned about that), but I cannot get the art package or the driver software that came with it to run through the PLUS D, though it will all function if the PLUS D is not

connected. Does anybody have a way of making these programs work with the PLUS D, or does it look like I might have had more than one duff chip? Is there a different driver program somewhere out there in the land for the AMX mouse? Any possibility of using it instead of a joy-stick?

I would very much appreciate any help with this, as I wanted to use the mouse for a lot of other things beside the art packages.

Thanks for a continually interesting magazine, it is a shame the Spectrum content is shrinking, especially when there is so much that could be done with interface designs (I even heard third hand, of hard drives being fitted to Spectrums by members on a continental user group!). Could yourselves or West Coast produce circuit diagrams, kits of parts etc. for things that might not be viable to manufacture in a costly production run but would still be worthwhile to us hobbyists?

Yours Sincerely, J.C.Blomley.

Can anyone throw some light on the mouse problems that Mr Blomley is having? *Ed.*

Dear Editor,

In Ray Bray's June Help Page he commented "I have always considered it a pity that the designers of the Z80 gave the programmer the ability to read all the registers but the program counter." He's quite right, at least as far as the Spectrum and the official Z80 Specs are concerned, but in fact there is a simple way to read the value of the program counter on SAM!

The trick is to use the instruction CALL 4 in your code. This returns the value of the program counter in the HL register pair. You can then add or subtract an offset (usually in DE) and perform 16 bit relative jumps - another feature normally missing from the Zilog repertoire. Even relocatable CALLs are

possible, by generating position-relative offsets to the required subroutine and the return point. These offsets can be worked out by your assembler when you generate the code.

At address 4, Dr Andy squeezed in a tiny subroutine, known to its friends as FOWIA - Find Out Where I Am. FOWIA fits between restart vectors at the start of SAMs ROM. It consists of two instructions: POP HL : JP (HL)

These are all you need to find the address of the next instruction. CALL 4 puts that address on the stack. The first line of FOWIA copies this to HL; the second jumps back to the line after the CALL. FOWIA does not appear in the Spectrum ROM, but you can get the same effect by storing the code at a known, static address in ZX RAM. There's a gap in the system variables at 23728 (\$5CB0) and you can use that, as long as no other program has got there first! For instance:-

```
1000 POKE 23728,225 :REM POP HL
1010 POKE 23729,233 :REM JP (HL)
```

These two lines set up FOWIA among the unused system variables. Subsequent machine code that wants to know its own address need only call 23728. This technique will work on all Z80-based computers, as long as you know of two bytes at fixed addresses that are available on every model.

FOWIA is potentially very powerful because it means you can write substantial Z80 programs that can run from any address without modification. This makes it much more likely that several programs will be able to co-exist on one machine.

EMULATION NEWS: I hear from William James that Speculator, his Amiga Spectrum Emulator, can now load ZX cassettes, as well as files from MGT discs and data transferred between serial ports. William's interface uses one of the Amiga's standard ports, so all you need

is a bit of wire and two plugs to get it working! Tape decoding is handled by one of the Amiga's on board timers, so the code runs reliably on all Amigas from the old A1000 and A500 to the latest A4000s, regardless of speed of the machine - although emulation after loading is still pretty sluggish on the old 16 bit models.

Amiga Speculator is still being tested - although several **FORMAT** readers have advance copies following the details you printed last year - and I shall write again when the full version is available for review. In the meantime, interested readers can find out more from William James, 54, Victor Road, Newtown, Colchester, Essex, CO1 2LX, UK.

PS. Very interested to hear of SAM FORTH, and the possibility of a Jupiter ACE emulator for Z80 micros (and impersonators!) more info please!

Yours Sincerely, Simon Goodwin.

Thanks for the info Simon. And if someone sends me more details on FORTH then I will be sure to pass it on. *Ed.*

Dear Editor,

I'm afraid I must take exception to what appears to be a very unpleasant attitude in answer to Bob Haslam's letter re different locations for fairs.

I am disabled and would also like venues nearer home sometimes, not a retort like that published implying buggar you-Jack, your requirements are not as important as the continentals.

Yours Sincerely, D.E.Rawlinson.

I'm sorry if my comments offended you. What I meant to get across is that wherever we hold the show some people are going to have to travel further than others. Ours is a very small country and if people can travel from as far as Germany to get to a Gloucester Show then I'm sure that nowhere in mainland Britain is out of reach. The shows have

to be held somewhere and Gloucester is as well served by the motorway network as most places. To move the shows to London or Birmingham or Manchester would add considerably to the costs, as would switching the venue each time we held a show. As you are only about 150 miles away I'm sure that you will be able to make it to Gloucester for one of the shows in future and we will look forward to seeing you. *Ed.*

Dear Editor,

Coo, lots of things to say. Firstly, big thanks to Bob Chowdhury for the info on *Chaos*, maybe soon I can kill my close friends in mortal combat again.

These Gloucesters shows. I want to go to one to look, maybe make some money with my games (Mungus software, plug, plug), but I never get enough notice from you beforehand, and the only way I can get up there is by train. Is there any easy way to get from the nearest station to the show?

On page 108 of the SAM User's Guide, it's got the diagram of all the bits and pieces you can attach to SAM. Do things like the scanner, digitiser and modem work? Can you get software for MIDI?

If you want more articles, do you need a games reviewer? I could do some reviews of ones already out, mostly Revelation ones, and new ones if I can afford them. I can also do little hints and things on how to use Gamesmaster if anyone's interested. Frank Evans (letter in Vol 8 No 11) had some good ideas for columns ('For Starters' and the skeleton programs), but you didn't seem to acknowledge them. I think they're great ideas because I can only do a little Basic, and use Gamesmaster for machine code. They'd be something I can really use. Excuse the bad spelling if any, my old PC has no spellchecker, and I don't have a printer for SAM. I don't talk this much in real life, congrats on **FORMAT**, it's a good magazine. Bye, and may the Lord

be with you.

Yours Sincerely, Stephen McGreal.

We always try to give at least three months notice of show dates Stephen, have you been skipping bits of **FORMAT**? And read this months editorial about lifts.

Most of the bits you mention were on Bruce Gordon's drawing board when the manual was written (September 1989) but have yet to appear. There is software for the MIDI but it has not been advertised for some time, try FRED Publishing as they have a good range of third-party software available.

As to games reviews. Yes, what I am looking for is lots of small reviews from readers covering their favourite game(s) on SAM or Spectrum. Just a few lines describing the game and saying what you like about it most. Give the games title, publisher and RRP together with ratings out of 10 for graphics, gameplay and overall value for money. We will then collate these mini-reviews together for a small supplement to go out in the New Year. Any articles you can write on GamesMaster would be welcome for future **FORMAT**s so get writing. *Ed.*

Dear Editor,

This is the very first time that I am writing to you with a very interesting idea and also, a problem to be solved, by some of the experts of our much loved Speccy.

I have a Spectrum 128K with PLUS D and 3 $\frac{1}{4}$ " dual drive and Epson Printer. Also, a Veloce Electronic Piano with Midi in, out, and through.

My question is, can I play the piano (I do) so my Spectrum will collect the music and do the following:-

1. Print, as I play, on screen the music on a proper sheet on the screen and then, allow me to adjust, delete, add, etc., on the screen this sound?

2. Once I am satisfied with the above results then, can I save to disc?

3. Can I send it to the printer to have a proper sheet of music?

4. Can I load from disc, alter date, to send it to the piano and played through it, also print again, if required?

5. What cables do I require?

6. What program do I require?

And the last (good he says)

7. Can anybody help?

I am sorry for the spelling but my English is a little limited.

P.S. Does anybody like Acorn BBC Model B? Offers please.

Hoping to read an answer in the best magazine in the World.

Yours Sincerely, L.Senna-Cheribbo.

Well I don't know of any MIDI software for the Speccy that will do what you want but I'm sure if there is such a thing one of our readers will let us in on the secret. As to your BBC, send more details of what you want to sell and we will run it in the small ads section. *Ed.*

Dear Editor,

A while ago you said that the mouse available from DATEL is not compatible with Art Studio. However, I have one of these mice/interfaces and use it with the Advanced Art Studio (Also available from DATEL). This program is 128K only and has a scrap-book, ram-disc support and other extra features, this version accepts Kempston and AMX mice. When I bought it I paid £5 extra for one with a through-port.

Yours Sincerely, David Powell.

I stand corrected David, many thanks for writing. *Ed.*

Letters may be shortened or edited to fit on these pages although we try to edit as little as possible.

This is YOUR letters page so it is up to you to fill it with interesting things. Come on, get writing, any subject even remotely related to computers. Just keep things as short as you can so we can fit in as many as possible each month. Please write clearly or type your letters. Send them to the address on page 3 or fax them to us on 0452 390890.

SOFTWARE

SAM Coupé Software On 3 $\frac{1}{2}$ " Disc

PCG's DTP PACK. The famous package including WordMaster, TypeLiner, HeadLiner. Complete on disc for SAM. R.R.P. £37.50. **SPECIAL READER'S BONUS** - Now includes 2 extra font packs worth £15.90 bringing the total pack value to £53.40.

Order Code: SCD-04 Price: £24.95

COLOUR DUMP. Screen dump software to work with Epson compatible colour printers. RRP £10.

Order Code: SCD-05 Price: £5.95

SAMSPEC SOFTWARE

SAM + Spectrum versions on 5 $\frac{1}{4}$ " disc (80 track 3 $\frac{1}{2}$ " or 5 $\frac{1}{4}$ " - please state on order)

FONT LIBRARY. 100 screen fonts plus support software, easy to use on either computer. Special +3 disc available (Spectrum files only). RRP £8.95

Order Code: SSD-01 Price: £5.95

MONEY MANAGER. Advanced personal budgeting system for both machines. Spectrum version needs 128K+2 (not +2a or +3). RRP £15.95

Order Code: SSD-02 Price: £9.95

SPECTRUM SOFTWARE

Unless otherwise indicated these are DISCIPLE/ PLUS D programs. Please state 3 $\frac{1}{2}$ " or 5 $\frac{1}{4}$ " disc (80 track only) on order.

ARTIST 2. The best art package ever written for the Spectrum. 48K and 128K versions on the same disc. With page-maker and utility software. Full manual. RRP £19.95

Order Code: SPD-01 Price: £14.95

UTILITY DISC #1. Six of the best and most useful disc utilities for DISCIPLE or PLUS D. Added bonus program - not to be missed. RRP £9.95

Order Code: SPD-02 Price: £6.95

PCG's DTP PACK. The famous package including WordMaster, TypeLiner, HeadLiner. R.R.P. £37.50. **SPECIAL READER'S BONUS** - Now includes 2 extra font packs worth £15.90 bringing the total pack value to £53.40.

Order Code: SPD-04 Price: £24.95

Special 3" disc version for +3

Order Code: SPD-04C Price: £29.95

FORMAT BACK-ISSUES

Volumes 1, 2 and 3 are no longer available. Volumes 4 and 5 of **FORMAT** are now available only as complete volumes.

Order Code: FBI-V4 (Vol.4) Price: £12.00

Order Code: FBI-V5 (Vol.5) Price: £12.00

All issues, from Volume 6 N° 1 to the latest issue, are available separately. To order just quote the Volume/Issue N°. **Price: £1.50 each, (5 or more £1.25 per copy).**

FORMAT

READERS SERVICES

FORMAT BINDERS

Hardcover binders to protect your valuable magazines. Each will hold 12 copies of **FORMAT** (or any other A5 magazines or manuals). Copies held in place by wires that just spring out for easy insertion or removal. Sorry, but we can't guarantee a specific colour (although we will try).

Order Code: FMB-01 Price: £4.95

HARDWARE

(For SPI please see West Coast advert.)

DUC LEAD For use with DISCIPLE or PLUS D. This 'Dueling Up Cable' allows two disc drives to be attached without the need for opening one drive to alter its ID. Allows you to switch drives to even out wear without opening cases. RRP £16.95

Order Code: DUC-01 Price: £11.95

PRINTER LEAD. Top quality parallel printer lead (BBC standard). 2 metres long. For use with DISCIPLE, PLUS D or SAM. RRP £10.95

Order Code: PPL-01 Price: £7.95

+3 PRINTER LEAD. For +3 or +2a. RRP £12.95

Order Code: PPL-02 Price: £9.95

SAM COUPÉ TECHNICAL MANUAL Version 3.

The real inside info on SAM. No SAM is complete without one. RRP £18.95

Order Code: STM-01 Price: £12.95

NEW NEW NEW NEW NEW NEW

Music Maestro For The +3

Supplied on 3" disc this excellent program allows you to write music either just for the fun of it or to include in your own games software where it plays under interrupts. Full manual, lots of demo tunes. The ideal way to write music. R.R.P. £9.95

Order Code: SPD-05C Price: £7.95

ORDERING All prices include UK p.p., overseas readers please add 10% to for extra postage.

Clearly state Order Code, Product description, Quantity required and Price. Remember to add any postage and don't forget your Membership Number or we can't process your order. Payment in STERLING by Cheque (drawn on a UK bank), P.O. Euro Cheque or Cash. Make cheques payable to **FORMAT**. Payment **MUST** be in same envelope as order. Send to **FORMAT's** address on page 3. Normally dispatch is of the same time as your next issue of **FORMAT**. We will not be held liable for delays or non-delivery due to circumstances beyond our control.